



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/735,482	12/12/2003	Georg Neumann	021756-003500US	2477
51206 7590 01/07/2009 TOWNSEND AND TOWNSEND AND CREW LLP TWO EMBARCADERO CENTER 8TH FLOOR SAN FRANCISCO, CA 94111-3834				
EXAMINER MITCHELL, JASON D				
ART UNIT 2193		PAPER NUMBER		
MAIL DATE 01/07/2009		DELIVERY MODE PAPER		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

10/735,482

Applicant(s)

NEUMANN ET AL.

Examiner

Jason Mitchell

Art Unit

2193

Period for Reply -- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 20 October 2008.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1, 4-43 and 45-50 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1, 4-43 and 45-50 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/S508)
Paper No(s)/Mail Date _____
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____

DETAILED ACTION

This action is in response to an amendment filed on 10/20/08.

Claims 1, 4-43, 45-50 are pending in this application.

Response to Arguments

Applicant's arguments filed 10/20/08 have been fully considered but they are not persuasive.

In the par. bridging pp. 2 and 3 the applicants state:

Applicants submit that Friedman discloses that the "JAVA programming language provides platform independence." (Freidman at col. 2, 11. 12-13). The Office Action further states that "[i]t would have been obvious to one of ordinary skill in the art at the time the invention was made to implement the functionality of the Java SE-Chen combination in the JAVA language." (Emphasis provided.) Applicants respectfully disagree with this assertion. As is known in the art, the JAVA programming language is written in ANSI C; nevertheless, Applicants respectfully submit that it would be impossible to implement Java SE in the JAVA language. In other words, the API (i.e., methods, systems, classes, etc.) of a language (i.e., Java SE) cannot be implemented using the same language. For example, just as the C programming language cannot be written using C, the JAVA programming language cannot be written using JAVA. Thus, for at least this reason, Applicants submit that Java SE in view of Chen, and further in view of Freidman, fails to make claims 1, 36, 38, 39, 45, and 46 obvious.

The examiner respectfully disagrees. First it is noted that the statement indicating "the JAVA programming language is written in ANSI C" is inexact. While JAVA *compilers (and/or java virtual machines etc.)* written in ANSI C were known in the art, the "JAVA programming language" itself is an abstract concept and can not properly be said to be written *in any* programming language. Additionally while it is reasonable to assume that the original JAVA compiler and VM were written in ANSI C, this is by no means a requirement and it is asserted that a reasonable search would produce both

compilers and VMs written in other languages. However no such search has been done here because, as discussed below, the rejection does not rely on such a teaching.

Further, the assertion that "it would be impossible to implement Java SE in the JAVA language" is incorrect. Those of ordinary skill in the art will recognize that the JAVA SE reference describes a collection of JAVA classes (i.e. classes written in the JAVA language). It is further noted that it is common place for the APIs for a language to be written in that language. Otherwise some additional interfacing technology (e.g. CORBA, XML transformation) would be required to make calls to the API.

Regardless, the applicants appear to be misrepresenting the rejection. The combination relied upon involves writing Chen's application (not the Java Language or a compiler) in JAVA and further making use of a JAVA class (described in the Java SE). Both of these actions were well within the abilities of those of ordinary skill in the art of software development. Accordingly, the applicants' arguments are not persuasive.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1, 4-18, 20-43 and 45-50 are rejected under 35 U.S.C. 103(a) as being unpatentable over "Java™ 2 Platform, Standard Edition v1.2.2 API Specification" (Java SE) in view of "Specification-based Testing for GUI-based Applications" by

Chen et al. (Chen) and further in view of US 7,171,588 to Friedman et al. (Friedman).

Regarding Claims 1, 36, 38-39, and 45: Java SE discloses:

determining a cursor position (Interface AccessibleComponent, pg. 1, `getAccessibleAt(Point p)`);

ascertaining, based on the cursor position, an accessibility context associated with the cursor position (Interface AccessibleComponent, pg. 1, `getAccessibleAt(Point p)`);

identifying a component by reference to the accessibility context (Class AccessibleContext pg. 3 `"getAccessibleComponent()"`), wherein the accessibility context has an accessibility role (Class AccessibleContext pg. 3 `"getAccessibleRole()"`) that defines a set of properties, including at least one program method, associated with the accessibility context (Class AccessibleContext pg. 3 `"getAccessibleAction()"`), wherein the identified component comprises the set of properties (e.g. Interface AccessibleComponent pg. 2 `"getFont ()"`; `"setFont()"`);

searching a component hierarchy for an object having an accessibility context matching the accessibility context (Class AccessibleContext pg. 3 `"getAccessibleComponent()"`); and

playing back an event (Interface AccessibleAction pg. 1 `"doAccessibleAction(int l)"`).

Java SE does not explicitly disclose recording and retrieving an identified accessibility context, prior to 'playing back' an event based on the identified, stored and retrieved context.

Chen teaches recording, in real time, an identified event (pg. 207, Section 2 "record all the interesting events ... records every point and click applied to the GUI application");

retrieving the stored event (pg. 207 "repeated with the recorded test scripts"; this requires retrieving the stored event); and

playing back the stored event (pg. 207 "automatically repeated with the recorded test scripts") for testing purposes (Title "Testing for GUI-Based Applications").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the 'Capture/Replay' testing methodology taught by Chen with the accessibility tools disclosed in Java SE. Those of ordinary skill in the art would have been motivated to do so in order to provide an environment to test accessible, GUI-based applications (Chen Abstract "automate effective testing for applications with complicated graphical user interactions."). Such a modification would have been within the ability of one of ordinary skill and would have produced only the expected results (i.e. a testing environment as in Chen accessing the GUI interface using the Java SE objects).

The Java SE-Chen combination does not explicitly teach operation independent of an operating system

Friedman teaches that Java executes independent of an the operating system (see e.g. col. 2, lines 12-16 "JAVA programming language ... provides "platform independence" ... an application is intended to perform the same regardless of the hardware and operating system on which it is operating").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to implement the functionality of the Java SE-Chen combination in the JAVA language. Specifically, by implementing the testing environment disclosed by Chen (pg. 207, section 2 "the Capture/Replay technique") in the JAVA language and making use of the Java SE object (e.g. Interface AccessibleComponent, pg. 1, getAccessibleAt(Point p)) to obtain the 'event' data to be recorded. Those of ordinary skill in the art would have been motivated to do so in order to achieve the platform independence provided by the JAVA language (Friedman col. 2, lines 12-16 "JAVA programming language ... provides "platform independence").

Regarding Claim 4: Java SE discloses the object comprises the set of properties, including the at least one program method, defined by the accessibility role (e.g. Interface AccessibleComponent pp 1-3, Method Summary).

Regarding Claims 5-12, 40-42: Java SE discloses responding to the claimed triggering events (e.g. Class Event and Class EventListener). Further in regard to claims 7 and 12, Java SE discloses executing program method a first or second time (Interface AccessibleAction pg. 1 "doAccessibleAction(int I)").

Regarding Claim 13-18 and 37: The claims recite recording and subsequently modifying records. Java SE does not explicitly disclose this.

Chen teaches recording GUI actions and modifying the created records (pg. 207, Section 2 "a Capture/Replay tool is used to record all the interesting events ... edit the recorded test script").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the 'Capture/Replay' testing methodology taught by Chen with the accessibility tools disclosed in Java SE in order to provide an environment to test accessible, GUI-based applications (Chen Abstract "how to automate effective testing for applications with complicated graphical user interactions.").

Regarding Claims 20-26: The claims recite various known GUI elements that Java SE discloses are implementable as AccessibleComponents (Interface AccessibleComponent pg. 1, "any object that is rendered on the screen").

Regarding Claim 27: Java SE disclose the action of formatting a portion of text (Interface AccessibleComponent, pg. 2 "setFont()")

Regarding Claims 28-30: The claims recite iterating over the steps of claim 1. Chen teaches applying the "Capture/Replay" tool to all actions (pg. 207, Section 2 "a Capture/Replay tool is used to record all the interesting events").

It would have been obvious to one of ordinary skill in the art at the time the invention was made to perform the steps of claim 1 iteratively in so as to "to record all the interesting events" (Chen pg. 207, Section 2).

Regarding Claims 31-35, 43: The claims recite standard testing and analysis functionality as taught by Chen. Further, regarding claim 35, it would have been obvious to test the application for accessibility as claimed (Java SE Interface AccessibleComponent "determine if an object supports the AccessibleComponent interface")

Regarding Claim 46: The claim recites the limitations of claim 1 except wherein the 'replaying' is analyzing the applications accessibility. This limitation has been addressed in the rejection of claim 35.

Regarding Claim 47: See the rejection of claim 12.

Regarding Claims 48-50: See the rejection of claims 31-35.

Claim 19 is rejected under 35 U.S.C. 103(a) as being unpatentable over “Java tm 2 Platform, Standard Edition v1.2.2 API Specification” (Java SE) in view of “Specification-based Testing for GUI-based Applications” by Chen et al. (Chen) in view of US 7,171,588 to Friedman et al. (Friedman) and further in view of US 2002/0188613 to Chakraborty et al. (Chakraborty).

Regarding Claim 19: The Java SE-Chen-Friedman combination does not explicitly disclose storing the records in an XML file.

Chakraborty teaches storing data in an XML file (par. [0019] “data is XML stored in flat files”)

It would have been obvious to one of ordinary skill in the art at the time the invention was made to store the records retrieved by the Java SE-Chen-Friedman system in XML as taught by Chakraborty (e.g. par. [0019]). Those of ordinary skill would have been motivated to do so in order to provide platform independence to the data storage mechanism (Chakraborty par. [0019] “XML stored in flat files ... helps the user read, edit and delete application data through a pure application program interface ... such as the Java API”)

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason Mitchell whose telephone number is (571)272-3728. The examiner can normally be reached on Monday-Thursday and alternate Fridays 7:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Bullock Lewis can be reached on (571) 272-3759. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Jason Mitchell/
Examiner, Art Unit 2193

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193